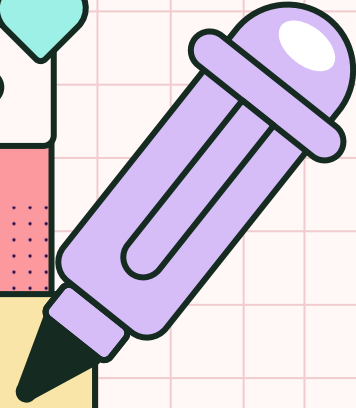




enactus[™]
University of Sheffield



CODE CREATORS
SHEFFIELD



2024 PYTHON BEGINNER COURSE

Lesson 0



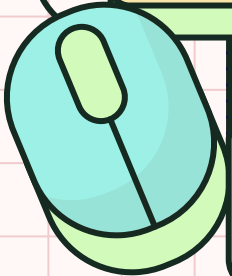
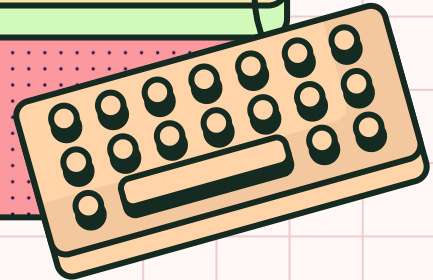
SHALEV LAU



/in/shalevl



/mushroomhater07



Introduction



- Introduce Enactus
- Social Media
- Why learn Python
- IDE selection
- Introduce Terminal
- VS Code Installation
- Python Installation
- Git Usage
- Generative AI

Question:



**DOES ANYONE KNOW
WHY WE CALLED
THIS LECTURE 0?**



enactus
University of Sheffield



**CODE CREATORS
SHEFFIELD**

Answer: Why Week 0



Arrays in Python (and most programming languages) are 0-indexed so you count from 0 instead of one.

I'm going to leave you on a cliff-hanger until we learn more about Python's data structures.

I believe we should walk before we can run, so we should learn the fundamentals first.



enactus
University of Sheffield



Thank you



University Life is busy.
We appreciate you spent your time wisely with us.
We hope you get the most out of this course.

By signing up for the course:

- You joined the community
- You get constant support from many other our experienced mentors
- You have help funded our social enterprises
 - OnTarget
 - CARTE
 - Vape Drop



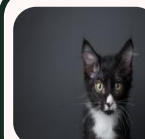
Meet The Team



Rikzar
Project Manager



Rokas
Deputy Proj. Manager



Julian
Financial



Amin
Marketing



Izenith
Outreach



Shalev
Program Developer



Nedjm
Program Developer



Haris
Mentor



Jivanthika
Mentor

Follow Our Social



Instagram
@code_creators

- Keep you updated
- quick DM reply with queries relating to finance
- See what goes on at CC



Discord Server
discord.gg/DX3S8EURz2

- Support from CC Mentors
- Private community for CodeCreators members



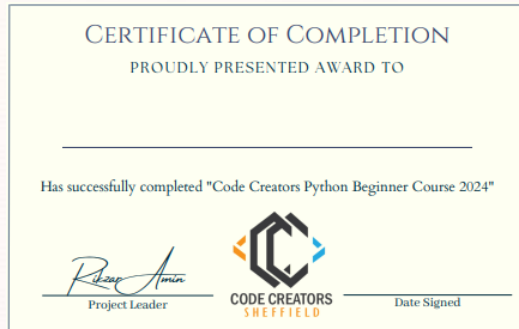
GitHub Resource
@ccsheff24

- Starter project files are here.
- Contribute to open source.

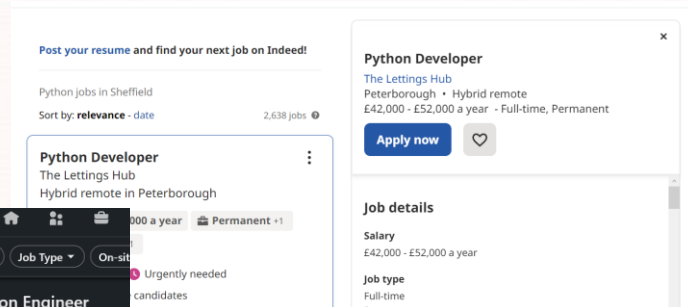
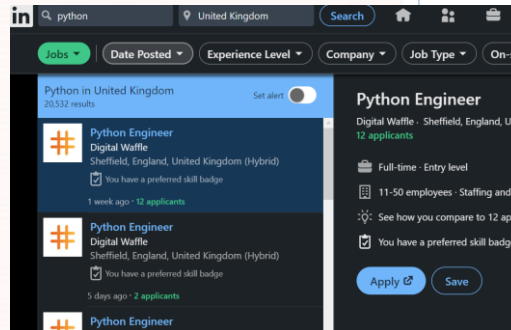
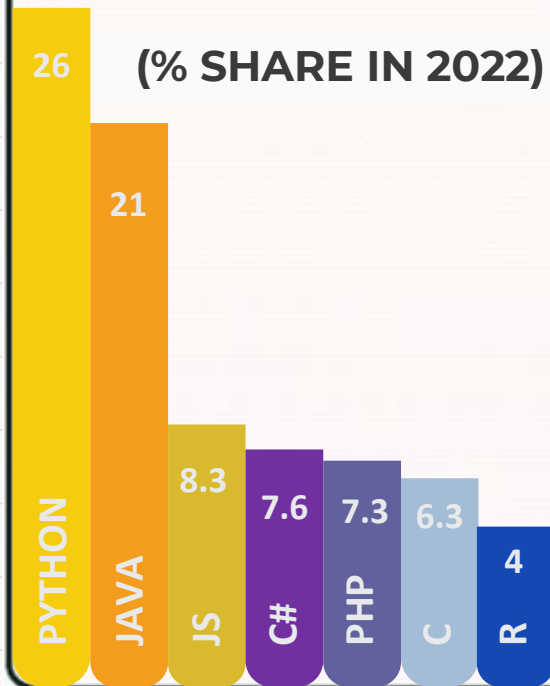
Recognising your effort



After you have completed our final project, you will be awarded with a digital website friendly version of certificate of completion tailored for you to upload on LinkedIn profile or your own CV.



Why learn Python: EVERGROWING POPULARITY



You won't be short of employment after learning Python!

Everybody in this country should learn to program a computer... because it teaches you how to think

Steve Jobs, co-founder and CEO of Apple Inc. (1955 - 2011)



Why learn Python: BEGINNER FRIENDLY



Python provides simple syntax to write your code. This is thanks to its indentation style, lack of triviality like terminating semi-colons or type definitions and a rich in-built standard library. Less time for errors, more time for coding!



PYTHON

```
# no import needed  
print("Code Creators!")
```



C

```
#include <iostream>  
using namespace std;  
int main(int argc, char * argv[])  
{  
    cout << "Hello, world!" << endl;  
    return 0;  
}
```



enactus
University of Sheffield



Why learn Python: RANGE OF APPLICATIONS



Web Dev (Covered in the advanced course.)

- Make websites using frameworks like Django.
- Ability to handle asynchronous requests.



Machine Learning

- Plenty of libraries for ML like Keras, TensorFlow, and Scikit-learn for machine learning.
- Python acts as an easier interface compared to Java etc.
- Python performs slower than C. Under the hood these libraries run on C anyway so it doesn't matter.



Big Data Analysis

- In-built features of supporting data processing for unstructured and unconventional data which is a common need in big data.
- Libraries like Pandas and SciPy aid with analysing data.

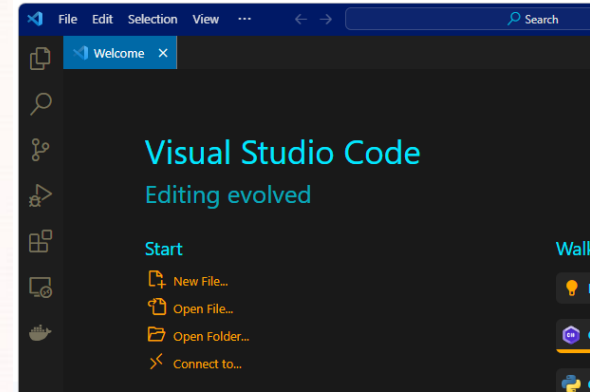
Code editor / IDE setup



Throughout the rest of the course, we will use **Visual Studio Code**. However, you are free to use any code editor or Python IDE you want.

Recommended:

- PyCharm
- VS code
- Thonny
- Atom
- IDLE (This comes with the express Python installation so no further setup is required. Simply search for "IDLE" in your start menu.)

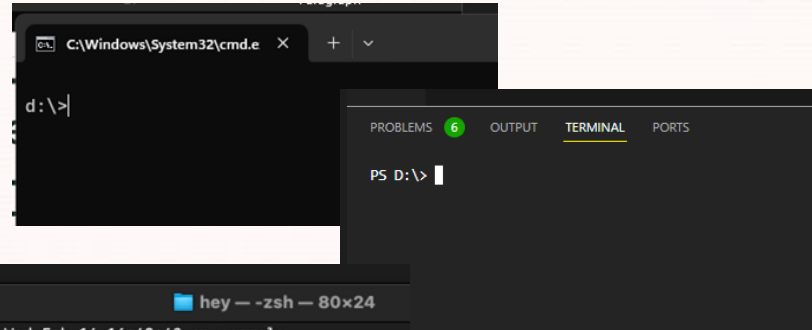


Terminal

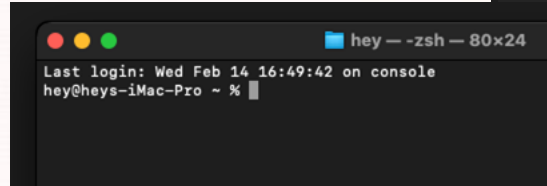


A window where user interact with the operating system
A text-based console application by entering text input through the computer keyboard and reading text output from the computer terminal.

Windows console is called the command prompt window & accepts MS-DOS commands.



macOS terminal accepts zsh commands, similar to sh command in Linux.

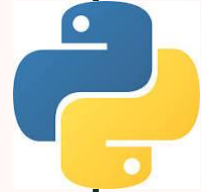


Python Installation



For Windows:

```
winget install -e -i --id=Python.Python.3.11 --source=winget --scope=machine
```



For macOS:

1. Open the Terminal app and install brew
`/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"`
2. Install python with: `brew install python`
3. Type `nano ~/.bash_profile` and press Enter.
4. Add the following line to the file:
`export PATH="$PATH:/Library/Frameworks/Python.framework/Versions/3.12/bin"`
5. Save the file by pressing **Ctrl+O**, then press **Enter**.
6. Close the file by pressing **Ctrl+X**.
7. Restart the Terminal app for the changes to take effect.

VS Code Installation and Extension Setup



enactus
University of Sheffield



For Windows:

```
winget install Microsoft.VisualStudioCode --override '/SILENT  
/mergetasks="!runcode,addcontextmenufiles,addcontextmenufolders"'
```

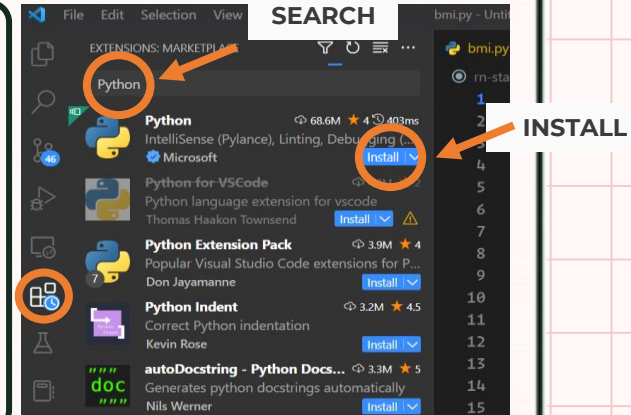
For macOS:

```
brew install --cask visual-studio-code
```

After Installation:

Click on the building blocks icon and search "Python" in the search bar and install the first extension that comes up. This gives us features like below and more. Feel free to add any other extension which you may find useful.

- IntelliSense (Pylance)
- Linting
- Debugging
- Code formatting



Course material download : GitHub Resource



For Windows:

```
cd %homepath%/Downloads
```

```
winget install --id Git.Git -e --source winget
```

For macOS:

```
cd ~/Downloads
```

1. Install Git with: `brew install git`

or

1. Install Xcode from AppStore (include Basic Git)

After Installation:

```
git clone <url>
```

For this week:

```
git clone https://github.com/ccsheff24/week1
```



Generative AI



Since the release of ChatGPT in November 2022, Generative AI has been optimized for generating code snippet and debug.

You are welcome to use Generative AI to encounter any coding problem. My advice is use it as your last resort as fixing errors trains your problem-solving skills. Also, you are welcome to ask our mentors on discord support channel

Some popular free Generative AI are:

- OpenAI's ChatGPT
- Google's Gemini
- Microsoft Copilot



Copilot

Your everyday AI companion



enactus
University of Sheffield

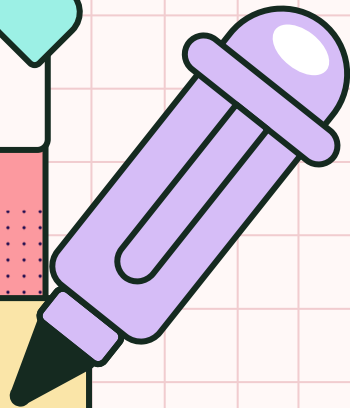




enactus
University of Sheffield



CODE CREATORS
SHEFFIELD



2024

PYTHON BEGINNER COURSE

Lesson 1



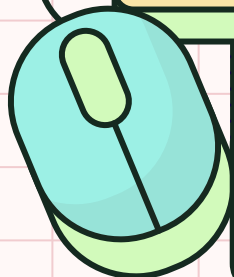
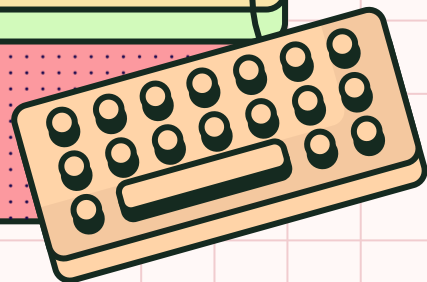
SHALEV LAU



/in/shalevl



/mushroomhater07



Python Basics



- Data Types
- Arithmetic operations
- Variable & constant
- Identifier
- Type Casting
- Input/ Output
- String operations
- Week 1 Project : BMI
- Advanced Project: °F to °C



enactus
University of Sheffield



Data Types

Data types define a possible set of values and what operations can be done on them. Typically languages have primitives (basic data types from which all other data types are constructed).

In the case of Python every value is an object but don't confuse this for an object type.

Data Types	Python Classes	Example	Description
Numeric	int, float, complex	20, 20.5, 2+5j	holds numeric values
String	str	"apple"	holds sequence of characters
Sequence	list, tuple, range	["apple", "banana"] ("apple", "banana")	holds collection of items
Mapping	dict	{"name": "John", "age": 36}	holds data in key-value pair form
Boolean	bool	True, False	holds either True or False
Set	set, frozenset	{"apple", "banana"}	hold collection of unique items
Binary	bytes, bytearray, memoryview	b"Hello"	Hold binary (non-printable) data

```
number = 20
fruit = "pear"
print(type(fruit))
# type <class 'string'>
```

Arithmetic operations



Python follows the traditional mathematical operations and rules of precedence (BIDMAS).

Operator	Operation	Example
+	Addition	<code>4 + 2 = 6</code>
-	Subtraction	<code>5 - 3 = 2</code>
*	Multiplication	<code>4 * 2 = 8</code>
**	Exponent	<code>4 ** 2 = 16</code>
/	Division	<code>7 / 2 = 3.5</code>
//	Floor (or Integer) division	<code>7 // 2 = 3</code> <i># 3.5 rounds to 3</i>
%	Modulus	<code>7 % 5 = 2</code>

If you ever need to do an operation on an existing variable you can put the operator in front of the assignment syntax.

```
x = 20
x = x / 2
print(x) #10
```

=

```
x = 20
x /= 2
print(x) #10
```



enactus
University of Sheffield



Variable



Variables work like they in maths, they store some value in memory.

What we just wrote is known by 3 names which are used interchangeably but incorrectly. Only one of them is technically correct in the context of Python.

- ~~Variable **declaration** – Declaration just specifies the type but doesn't allocate memory.~~
- ~~Variable **definition** – Definition associates the variable with a type and allocates memory.~~
- Variable **assignment** – The name on the left-hand side now refers to the result of evaluating the right-hand side, regardless of what it referred to before (if anything)

The name given to the variable is called Identifier

```
fav_website = 'apple.com'  
fish = 2
```



enactus
University of Sheffield



Python Identifier



The name that given to a variable and function are named as identifier

There are special rules:

- They cant be python keywords (**True, False, None, print**)
- They are case-sensitive (**Animal** & **animal** are different variable
- Whitespace is not allow (**calculated answer** → **calculated_answer**)
- Cannot use special symbol (e.g. **!, @, #, \$**)
- Begin with letter or **_** .Cannot start with digit (**1name**)




Good

Name
is_complete
num_of_people



Bad

lMAao!

1st_variable



enactus
University of Sheffield



Type Casting



Unlike other languages like JavaScript, Python prefer *explicit is better than implicit*. That means Python won't automatically type casting.

We also have to (explicitly) convert types so that they're compatible with each other.

```
num1 = int("5")
num2 = int(-2.8)
num3 = float("5")
```

```
print(num1) # prints 5
print(num2) # prints -2
print(num3) # prints 5.0
```

```
print("It's " + 11 + "AM")
# TypeError: can only concatenate str (not "int") to str
print("It's " + str(11) + "AM")
# It's 11 AM
```



enactus
University of Sheffield



Python I/O



We can use **print** keyword to output to terminal and get user input using **input** to assign a variable

print statement will automatically create a newline. To avoid this, we can add an argument **end = ''** before the closing bracket

input statement pauses execution of the program until **Enter** key is pressed. It takes 1 optional argument for the prompt

```
print("Is GTA", 6, "released?", False)
# Is GTA 6 released? False
```

```
print("hello", end = '')
print("world")
# helloworld
#
```

```
name = input("Enter your name: ")
print(name)
# Enter your name: James
# James
```

Format String

Sometimes, we don't want to spamming `,` or `+` while printing

`.format` is method available on all strings using the dot notation. If you specify index numbers, they'll correspond with the ordered value you supplied in the method. Or you can use keyword arguments. Otherwise, they'll follow consecutive order.

OR

use the format specifiers and the `%` operator. The format specifiers must be in order with the argument list and must match the same data type

`(%s` string `%f` float `%d` integer)

`f"string{variable}string"` you can embed variables and expressions directly in f-strings.

```
print('{2} {0} {1}'.format('are in',  
                             'order', 'We'))
```

```
# We are in order
```

```
print("%d %s sells %.1f dollars" %  
      (100, "apples", 48.23))
```

```
# 100 apples sells 48.2 dollars
```

```
last_name = "Obama"
```

```
print(f"His is {last_name}.")
```

THIS IS NOT THE SAME AS
THE MODULUS OPERATOR.



enactus
University of Sheffield



String Operation: substring



Strings are simply a sequence of character(s) that can form a word or sentence and an index is the relative location of a thing

In python, index starts counting from `0`. And use negative indexes to reference items at the end of the string. `-1` represents the last character and so on...

We can get a character from a long string. Stating the index we want inside `[]` bracket

We can also get a part of a long string. This is called substring. We use `:` inside the `[]` bracket.

If `[2:5]` is set, python will output from index `2` to `4` only

This logic also applies for lists. In fact, string is just a list of character

```
print("CodeCreators"[0]) # C
print("CodeCreators"[-2]) # r
# get index 2 and 3 only
print("CodeCreators"[2:4]) # de
```



enactus
University of Sheffield



Comment



You might notice hash(#) symbol in the previous example. In python, it is for writing a single-line comment.

Comment can be used to

- Explain code
- Make code more readable
- Prevent execution when testing

You can also write multiline comment by (`"""`). This usually appear at the front

```
"""  
Written on 2024  
Version 1.0  
"""  
  
fruit = "apple"  
# assignment used
```

Shortcut key for most text editor:

Ctrl + / if you are on Windows

Cmd + / if you are on a Mac.

Week 1 Project: BMI calculator



BMI is a way to figure out your approximate level of body fat in an easy, inexpensive way.

Requirement:

- user able to input height (in centimeters)
- user able to input the weight (in kg)
- user able to see the BMI index
- BMI index output with any string formatting

BODY MASS INDEX FORMULA (Metric)

$$\text{BMI} = \frac{\text{Weight (kgs)}}{[\text{Height (m)}]^2}$$

Final Result:

BMI Calculator

Enter your height in cm: 180

Enter your weight in kg: 75

Your BMI is: 23.148148148145

Week 1 Project: Advanced project



FAHRENHEIT → CELSIUS

USE THE FOLLOWING CONVERSION FORMULA:

$$^{\circ}\text{C} = \frac{5}{9} (^{\circ}\text{F} - 32)$$



Where:

C = degrees Celsius

F = degrees Fahrenheit

Final Result:

Temperator Convertor

Enter temperature in F: 59

Current temperature in C: 15



enactus
University of Sheffield



Meme of the week



Python is the
easier language
to learn.
No brackets,
no main.



You get errors
for writing an
extra space

